



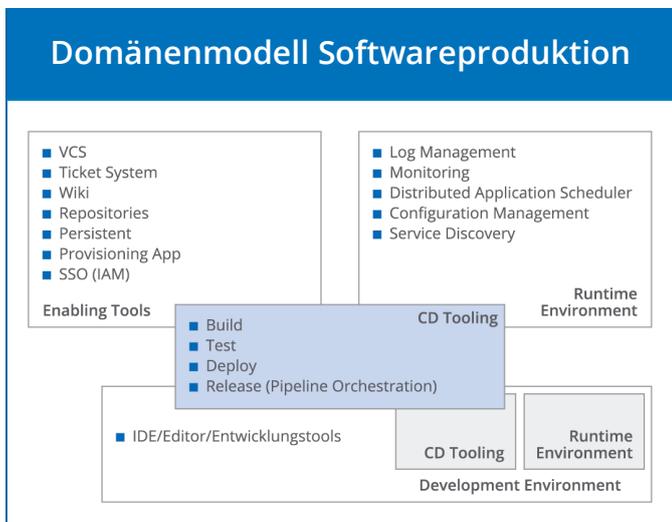
DevOps4U: Mehr Zeit für Features

Sie entwickeln die Software und generieren Mehrwert, wir managen Prozesse und Plattform

Moderne Softwareproduktion ist heute mehr als Continuous Integration. Regelmäßig Software zu bauen und mittels Unit- und einfacher Integrationstests zu testen, ist heute schon Standard bei vielen Entwicklungsteams. Dies reicht aber für den nächsten Schritt im Rahmen einer automatisierten Produktionskette nicht aus. Auch recht einfache Anwendungen bestehen heute aus mehreren Deployment-Einheiten, etwa einem JavaScript-basierten Frontend und einem Java Backend. Automatisierte End-to-End-Tests zur Sicherstellung der Releasefähigkeit werden häufig ergänzt. Das Ganze passiert nicht als Selbstzweck. So wie agile Entwicklungspraktiken den Feedbackzyklus für Entwicklungsteams reduzieren, wird mittels Continuous Delivery der Feedbackzyklus für ihre Fachbereiche verkürzt. Dazu müssen neue Features nicht nur schneller entwickelt, sondern auch schneller in Produktion gebracht werden. Unsere Experten unterstützen Sie dabei.

Architektur Ihrer Umgebung

Wenn wir die Softwareproduktion, also den Prozess von Bau und Test hin zur Auslieferung der Software, als Fachdomäne betrachten, dann haben wir es mit vielen verschiedenen Aspekten zu tun:



Sie nutzen bereits Continuous Integration?

Wenn Sie heute schon Continuous Integration nutzen, dann haben Sie wahrscheinlich schon viele Themen in der Domäne „Enabling Tools“ adressiert, wie Wiki oder Source Code Repositories.

Häufig findet man auch eine gewisse Integration dieser Tools, etwa über Web Hooks, um etwa auf Basis von neuem Code den Continuous Integration Server für Übersetzung und Test der Software anzustoßen.

Sie möchten Cloud-basierte Infrastrukturen nutzen?

Wenn Sie sich in Richtung von Cloud-basierten Infrastrukturen bewegen, dann kommen viele neue Fragestellungen auf Sie zu. Etwa wie Sie Ihre Entwicklungs- und Testumgebungen handhaben. Dazu gehört zum Beispiel das automatische Ausrollen von neuen und geänderten Umgebungen und deren Überwachung. Hierbei handelt es sich um Themen aus den Bereichen „Runtime Environment“ und Continuous Delivery Tooling.

Unbeliebt aber unabdingbar: Kontrolle und Optimierung

Tätigkeiten in diesen Bereichen werden von Entwicklungsteams häufig als lästig empfunden. Oft setzen sie sich nur einmalig zu Beginn eines Projekts intensiv mit dem Thema auseinander. Es fehlt den Teams daher an Routine. Aber gerade, wenn man Continuous Delivery als eine Erweiterung der agilen Entwicklungspraktiken versteht, braucht es eine regelmäßige kritische Betrachtung und Optimierung der Continuous-Delivery-Prozesse. Um Verbesserungspotenziale erkennen und umsetzen zu können und die Fähigkeit einzelne Technologien zu einem guten Continuous-Delivery-Prozess kombinieren zu können, ist eine fortlaufende Beobachtung der Wissens- und Werkzeugentwicklung im Continuous-Delivery- und Cloud-Umfeld unbedingt notwendig.

